

Aplicando Linguagem Natural ao Gerenciamento de Redes de Computadores através do Chatterbot Doroty

Michelle Denise Leonhardt¹, Liane Margarida Rockenbach Tarouco¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{mdleonhardt, liane}@inf.ufrgs.br

Abstract. *This work presents an investigation of the use of chatterbots as a network management tool, beyond a tool for network management training. A chatterbot is a program that attempts to simulate typed conversation in natural language, with the aim of talking as close as possible like a human being. In our approach, we have created Doroty, a network management-aware chatterbot that is able to interact not only with the network administrator, but also with the managed network via management protocols such as SNMP.*

Resumo. *Este artigo apresenta uma investigação do uso de chatterbots como uma ferramenta de gerência, além de uma ferramenta de capacitação profissional. Um chatterbot é um programa que procura simular uma conversação, em linguagem natural, com o objetivo tornar esta comunicação o mais próxima possível da conversação de um ser humano. Desta forma, foi desenvolvido Doroty, um chatterbot que é capaz de interagir não somente com o administrador de uma rede, mas também com a própria rede gerenciada através da utilização do protocolo SNMP.*

1. Introdução

No gerenciamento de redes de computadores, possíveis falhas ou comportamentos incoerentes devem ser diagnosticados e solucionados automaticamente ou através de um gerente preparado para tomar as providências necessárias [Bieszcad et al., 1998]. Essa atividade exige dos gerentes de redes a disponibilidade de uma grande quantidade de informações sobre os seus equipamentos, as tecnologias envolvidas e os problemas associados a elas.

Atualmente, cada vez mais as tecnologias de Inteligência Artificial (IA) vêm sendo utilizadas no domínio das redes de computadores e, principalmente, no gerenciamento de redes. Abordagens utilizando Sistemas MultiAgentes [Cheikhrouhou et al., 1998] e sistemas especialistas [Melchioris and Tarouco, 1999] já foram utilizadas com o objetivo de facilitar a tarefa de gerenciamento de rede aos olhos do usuário e de empregar novos paradigmas de programação para a tarefa.

Este artigo investiga o uso de *chatterbots* como uma ferramenta de gerência, além de uma ferramenta de capacitação profissional. Um *chatterbot* (também denominado no decorrer do artigo como *bot*) é um programa que procura simular uma conversação, em linguagem natural, com o objetivo tornar esta comunicação o mais próxima possível da conversação de um ser humano. Os *chatterbots* representam um grande potencial para a tarefa de treinamento e gerenciamento de redes já que incorporam novos conceitos e introduzem técnicas oriundas de outras áreas da Ciência da Computação, como a utilização da linguagem natural, campo de estudo fortemente explorado pela IA. Desta forma, foi desenvolvido Doroty, um *chatterbot* que é capaz de interagir não somente com o administrador de uma rede, mas também com a própria rede gerenciada através da utilização do protocolo SNMP (*Simple Network Management Protocol*).

3. O Chatterbot Doroty

A arquitetura de Doroty consiste dos seguintes elementos: a interface do *chatterbot* e sua base de conhecimento, o banco de dados de informações, banco de dados histórico, o módulo coletor, o módulo de atualização do banco de dados histórico e o módulo central, que armazena as regras de funcionamento do *bot*. O processo básico de funcionamento do *chatterbot* engloba algumas etapas. Um usuário interage com Doroty através de sua interface. A interface é constituída de um formulário para entrada da pergunta e uma área destinada para as respostas. Assim, o usuário faz uma pergunta ao *chatterbot* simplesmente digitando a pergunta em linguagem natural no formulário. O *chatterbot* processa essa entrada e procura a informação correspondente na sua base de conhecimento estática. Se a resposta não necessita explicação detalhada, o *bot* apenas fornece uma resposta teórica sobre o assunto. Por outro lado, nos casos em que uma explicação detalhada ou uma consulta nas condições da rede se fizer necessária, o *bot* ativa o módulo central e fornece, através de suas regras internas e informações em seu banco de histórico de dados, uma resposta dinâmica. (figura 1)

Para a construção de Doroty, foi utilizado como base o *chatterbot* ALICE [Wallace, 1995], um *chatterbot* amplamente utilizado na atualidade e que apresenta uma grande possibilidade de recursos. O software ALICE é gratuito, está disponível em diversas plataformas e apresenta grande facilidade de uso, apresentando a linguagem AIML e desenvolvendo continuamente melhorias em seu funcionamento. A arquitetura original de ALICE foi expandida, através da criação de módulos cujo objetivo é adaptar o comportamento do *chatterbot* de forma a permitir interação com a rede gerenciada.

3.1. Processamento interno

A base de conhecimentos de Doroty é construída usando AIML (*Artificial Intelligence Markup Language*) [Taylor, 2003], uma linguagem de marcação baseada no XML que, através de seus objetos e categorias, descreve o conjunto de padrões que o *chatterbot* reconhece. O AIML é baseado em padrões de entrada do usuário, conhecidos como categorias. Uma frase escrita por um usuário é comparada aos padrões descritos na linguagem e, com base neste processo, são selecionadas e construídas as respostas. O padrão do AIML é bastante simples e consiste apenas de palavras, espaços, e os símbolos especiais `_` e `*`. As palavras podem ser formadas por letras e numerais e são separadas por um espaço simples. Os caracteres especiais `_` e `*` funcionam como frases ou palavras.

No caso de Doroty, a programação dos arquivos AIML explora muitos dos recursos da linguagem. Isso porque o *chatterbot* deve ser preparado para responder também perguntas dinâmicas, ou seja, valores devem ser passados para o módulo central a fim de se buscar alguma informação do estado atual ou anterior da rede gerenciada. Para que esses valores sejam passados corretamente, foi definido um conjunto de parâmetros, que são atingidos pelo AIML através de comandos de recursividade e substituição de caracteres especiais de entrada. Esses parâmetros são: tópico, endereço, operação, tempo e situação. Os parâmetros são setados de acordo com cada pergunta feita pelo usuário pelo próprio programador do AIML. Depois de setados tais parâmetros, o AIML dispara a execução do módulo central.

O módulo central é responsável por receber toda a informação gerada e processar essa informação de modo a enriquecer as respostas de uma maneira mais prática. Além disso, também é o módulo responsável por receber os parâmetros e aplicar regras de tratamento para cada combinação de entrada possível. Entende-se por combinações de entrada o conjunto de parâmetros que são efetivamente passados, desconsiderando os que têm valor nulo.

Cada um desses casos apresenta um conjunto de regras de tratamento próprias e vai buscando as informações do banco de dados da forma mais adequada. O banco de dados de Doroty é constituído por 7 tabelas que permitem identificar uma rede, seus equipamentos e o tipo, os objetos que podem ser consultados e suas características, o histórico de informações da rede e

outras informações que permitem associar cada tópico a um ou mais objetos. Ele está estruturado de forma a permitir que sejam cadastradas informações sobre mais de uma rede, além de conter informações detalhadas sobre cada objeto da rede.

A tabela Histórico serve para armazenar medições da rede que vão sendo feitas em determinados intervalos. O intervalo pode variar para cada objeto e é armazenado juntamente com as informações do objeto dentro do banco. A atualização dessa tabela é constantemente feita pelo módulo histórico. O módulo histórico é programado para percorrer, a cada minuto, a tabela de histórico de dados em busca da última atualização do valor de cada objeto. Se a última atualização de um objeto tiver ocorrido em um tempo maior que o intervalo de atualização desejado, o módulo coletor é chamado para que aquele objeto seja novamente consultado.

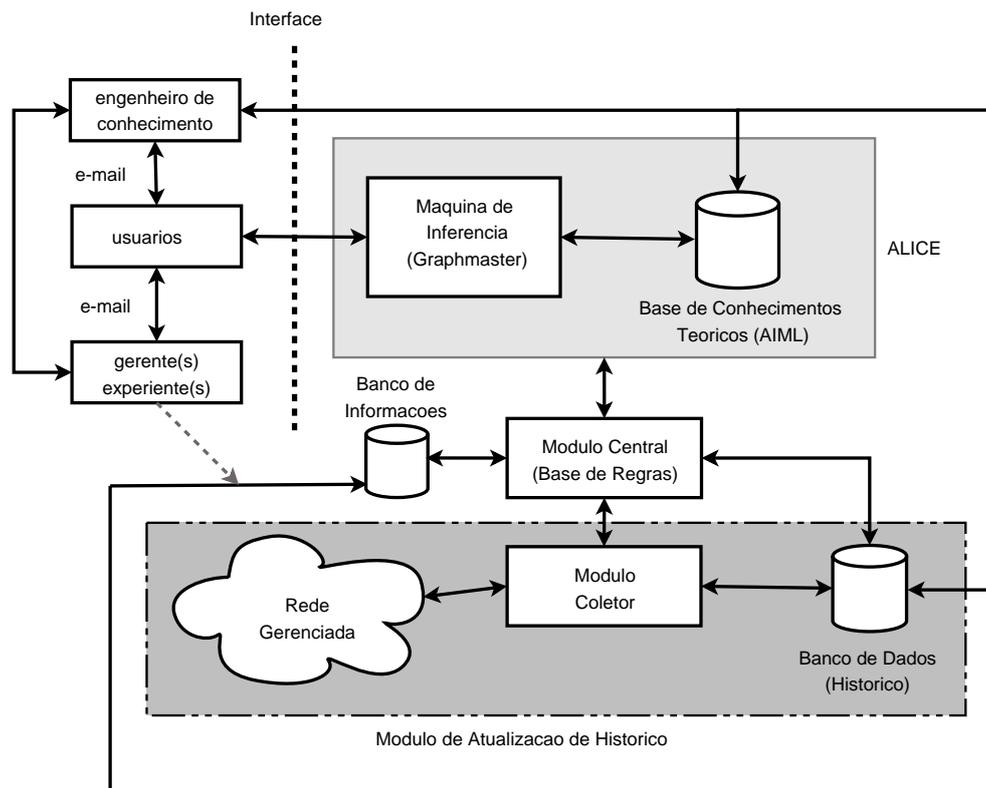


Figura 1. Diagrama detalhado da arquitetura de Doroty (Interação)

Outro módulo importante na implementação de Doroty é o módulo coletor, responsável por coletar as informações da rede gerenciada. Utilizando o protocolo SNMP, os valores de cada objeto podem ser capturados com o objetivo de permitir que o usuário obtenha o estado atual de sua rede. Embora o módulo seja programado para executar comandos *get*, *get-next*, *get-table* e *set*, este último não é habilitado já que o público alvo de Doroty é formado por administradores de rede em treinamento, ou, por consequência, mais aptos a cometer algum erro na configuração de objetos.

Cada tipo de pergunta é associado a um caso. A todos os casos devem ser atribuídos tópicos para que o *chatbot* possa se direcionar para o assunto desejado. Quando as perguntas buscam uma informação atual da rede, sem que haja necessidade de consulta ao banco histórico, os parâmetros de tempo e situação são setados para nulos. Já os casos em que há uma consulta ao banco histórico, o parâmetro tempo passa a receber o valor desejado. Quando ocorre uma situação especial em que o usuário faz uma pergunta incompleta, sem passar a informação sobre qual equipamento deseja consultar, o módulo central se encarrega de percorrer o banco de dados verificando quais equipamentos podem conter a informação desejada e mostrando ao usuário uma

lista de endereços possíveis para a consulta, permitindo com que o usuário refaça a pergunta de forma completa, simplesmente escolhendo entre um dos equipamentos.

Ao final do processamento de cada tipo de pergunta, Doroty formula uma resposta, baseada no objeto consultado e nas características da pergunta do usuário. Essas respostas são programadas para, além de fornecer o estado atual da rede, ensinar ao usuário como proceder para encontrar a informação sozinho. Assim, ela adiciona em suas respostas *links* para tutoriais sobre o assunto, fotos, animações ou bibliografia. Dessa forma, o sistema busca atingir seu grande foco: o de treinar usuários até que se sintam confiantes para administrar a rede.

5. Conclusões e Trabalhos Futuros

As redes de computadores estão se tornando maiores e mais heterogêneas, de modo que constituem um domínio em que o uso de tecnologias de IA proporcionam inúmeros benefícios. Administradores de sistemas e de redes podem utilizar tecnologias de IA para executar processos críticos na área de gerenciamento de redes, como, por exemplo, detecção e correção de falhas e monitoração de performance.

Em se tratando da aceitação de uma tecnologia cujo principal atrativo é o uso da linguagem natural e a conseqüente facilidade de interação com o usuário, foi constatado que o uso da solução representa um enorme potencial no sentido de aumentar o interesse e curiosidade dos usuários. Por imitar a realidade e ser de fácil uso, o bot permite que uma pessoa com menos experiência não se sinta constrangida em interagir com ele, como muitas vezes acontece em ambientes profissionais e acadêmicos com os chefes e professores, respectivamente. Esse resultado foi ao encontro de resultados de outros experimentos, nas mais diversas áreas, que utilizaram *chatterbots* para interação com usuários em seus campos de domínio [Bickmore, 1999] [Rickenberg and Reeves, 2000].

Alguns trabalhos que podem agregar valor ao *chatterbot* desenvolvido englobam o desenvolvimento de um módulo para criação automática de AIML, baseado em extração de conhecimento de textos, livros e artigos da área; o desenvolvimento de um módulo para que o usuário possa avaliar a qualidade de cada resposta e sugerir novos recursos; a transformação do sistema em uma ferramenta completa de auxílio ao gerenciamento de redes através da integração de um módulo responsável por realizar a verificação e correlação de alarmes de acontecimentos em uma rede. Dessa forma, o *chatterbot* poderia desempenhar um papel mais ativo, antecipando as informações relevantes a um usuário.

Referências

- Bickmore, T. W. (1999). Social intelligence in conversational computer agents. Proseminar conceptual analysis of thesis area, Massachusetts Institute of Technology, Cambridge, USA.
- Bieszcad, A., Pagurek, B., and White, T. (1998). Mobile agents for network management. IEEE Communications Surveys, 1(1).
- Cheikhrouhou, M. M., Conti, P., and Labetoulle, J. (1998). Intelligent agents in network management a state-of-the-art. Networking and Information Systems, 1(1):1–29.
- Melchior, C. and Tarouco, L. M. R. (1999). Fault management in computer networks using case-based reasoning: Dumbo system. In Proc. of the International Conference On Case- Based Reasoning, Lecture Notes in Computer Science, pages 510–524, Seeon Monastery, Germany. Berlin, Springer.
- Rickenberg, R. and Reeves, B. (2000). The effects of animated characters on anxiety, task performance, and evaluations of user interfaces. CHI Letters, 2(1):49–56.
- Taylor, A. (2003). The aiml mini reference and primer. Disponível em: <http://hippie.alicebot.com/ataylor/index.html>.
- Wallace, R. (1995). Alice - artificial linguistic internet computer entity - the ALICE AI. foundation. Disponível em: <http://alicebot.org>